**1. Introduction:**

The Princeton Hospital is a [health care](https://en.wikipedia.org/wiki/Health_care) institution providing [patient](https://en.wikipedia.org/wiki/Patient) treatment with specialized medical and nursing staff and medical equipment.   
Princeton Hospital plans to develop a Case Report Generating application, A desktop application [Core Java Batches – Swing Application] where the receptionist will fill in the form details for a patient and a report will be generated by the patient for the particular doctor.

* 1. **Scope and Overview:**

This application allows doctors, admin and receptionist to create, read, update, delete about various information related to patients and hospital.

**2. System Overview**

The Case Report Form Generation should support basic functionalities for below-listed users.

* Administrator(A)
* Receptionist(R)
* Doctors (D)

**2.1 Authentication & Authorization**

**2.1.1 Authentication**

Any end-user should be authenticated using a unique login ID and

password.

**2.1.2 Authorisation**

The operations- supported and allowed would be based on the user

type.

**2.1.3 Environment**

* Hardware machine(PC AMD-2GHZ,512MB RAM,500 GB HDD)
* WINDOWS OPERATING SYSTEM
* Database-Oracle 6g or higher
* JRE 1.6 or higher
* Eclipse IDE

**3. Sub-system Details**

The patient report generation is defined with three types of users (Administrator, customer, receptionist) wherein all the users need to log in successfully before performing any of the respective operations.

Find below tables that provide functionality descriptions for each type of users/sub-system. Against each requirement, indicate data is listed in column “Data to ”.

**3.1 ADMINISTRATOR:**

The administrator as a user is defined to perform below-listed operations after successful login.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Objects | Operations | Data to Include | Remarks |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**3.2 RECEPTIONIST:**

The Receptionist as a user is defined to perform below-listed operations after successful login.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Objects | Operations | Data to Include | Remarks |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**3.3 DOCTOR:**

The Doctor as a user is defined to perform below-listed operations after successful login.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Objects | Operations | Data to Include | Remarks |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**4. DATA Organisation**

**4.1 Table: User Profile**

The user specific details such as name, address, authentication and authorization/ privileges should be kept in one or more tables, as necessary and applicable.

|  |  |
| --- | --- |
| Field Name | Description |
| User ID | Patient ID is auto-generated after registration and It is used as LoginID |
| Name | Customer Name |
| DOB | DOB of Patient |
| Gender | Gender of user |
| Present Address | Present address of patient |
| Permanent Address | Permanent address of patient |
| Phone Number | Phone number of patient |
| EmailID | Email Id of patient |

**5. Assumptions**

* UI: The type of client interface to be supported – GUI based.
* The scope of the application is limited to hospitals which avail only cabs.
* CRUD will be done by the admin.
* Patients will be admitted on the FIFO basis.
* Allotment of the room will be performed by Receptionist.
* The patient can choose amongst the available rooms.

6. General Expectations

* The server is a concurrent server servicing multiple clients.
* The database can be implemented using Oracle 6G or above.
* The application will support at least 1 admin 1 Doctor and 1 Receptionist.
* Compilation and build should be done using eclipse IDE.
* Source code and all documents must be maintained in configuration management system.
* Company coding standard must be followed.
* Deliverables should include compiled and tested source code, unit test code, wiper style report and system test plan/ report documents.

NOTE:

1. Validation of user data
2. User Interface Design.
3. Create at least one SQL DML Statement inside PL/SQL blocks.

7. Acceptance Criteria

All P1 requirements have to be mandatorily implemented.

8. Traceability to requirements.

Appropriate requirements from RS and FS are mapped here.